

```

# -*- coding: utf-8 -*-
bam = ("GGATCC", "CCTAGG", "BamHI")
hind = ("AAGCTT", "TTCGAA", "HindIII")
apa = ("GGGCCC", "CCCGGG", "ApaI")
targets = []
keys = {}
def window_score(a,b):
    s = 0
    locs = []
    for n,(i,j) in enumerate(zip(a,b)):
        if i!=j and i!="n":
            s = s + 1.5
            locs.append(n)
    s = s + max(locs) - min(locs)
    return s
for i,j in ((bam, hind), (bam, apa), (hind, apa)):
    targets += [i[0]+"nn"+j[0], j[0]+"nn"+i[0]] #, i[1]+"nn"+j[1]] #
    i[0]+"nn"+j[1], i[1]+"nn"+j[0],
    keys[i[0]+"nn"+j[0]] = i[2]+"-"+j[2]
    keys[j[0]+"nn"+i[0]] = j[2]+"-"+i[2]
    #keys[i[0]+"nn"+j[1]] = i[2]+"-"+j[2]
    #keys[i[1]+"nn"+j[0]] = i[2]+"-"+j[2]
    #keys[i[1]+"nn"+j[1]] = i[2]+"-"+j[2]
base1 =
["tttagcttccttagctcctgaaaatctcgataactcaaaaaatacgcccggtagtgatcttatttc
attatggtgaaagttggaacctcttacgtgcccgatcaactcgagtgccacctgacgtctaagaacc
attattatcatgacattaacctataaaaaataggcgtatcacgaggcaga".upper(),
    "gaagatcctttgatcttttctacggggtctgacgctcagtggaaacgaaaactcacgttaa
gggattttggtcatgagattatcaaaaaggatcttcacctagatccttttaaattaaaaatgaagttt
taaatcaatctaaagtatatatgagtaaacttggtctgacagctcgaggcttgg".upper(),
    "ttacttcgcggttatgcaggcttcctcgctcactgactcgctgcgctcggtcggtcggtcg
cggcgagcggtatcagctcactcaaaggcggttaatacgggttatccacagaatcaggggataacgcagg
aaagaacatgtgagcaaaaggccagcaaaaggccaggaaccgtaaaaaggccgcgttgctggcggtt".
upper()]
base2 =
["ttacttcgcggttatgcaggcttcctcgctcactgactcgctgcgctcggtcggtcggtcggtcg
gcggtatcagctcactcaaaggcggttaatacgggttatccacagaatcaggggataacgcaggaaagaa
catgtgagcaaaaggccagcaaaaggccaggaaccgtaaaaaggccgcgttgctggcggtt".upper(
),
    "gaagatcctttgatcttttctacggggtctgacgctcagtggaaacgaaaactcacgtta
agggattttggtcatgagattatcaaaaaggatcttcacctagatccttttaaattaaaaatgaagtt
ttaaatcaatctaaagtatatatgagtaaacttggtctgacag".upper(),
    "cacatttccccgaaaagtgccacctgacgtctaagaaaccattattatcatgacattaa
cctataaaaaataggcgtatcacgaggcaga".upper()]
results1 = []
for b in base1:
    hits = {}
    minimum = {'BamHI-HindIII':40, 'HindIII-ApaI':40, 'BamHI-
ApaI':40, 'HindIII-BamHI':40, 'ApaI-HindIII':40, 'ApaI-BamHI':40}
    for i,l in enumerate(b[0:-13]):
        for t in targets:
            score = window_score(t, b[i:i+14])
            if score < minimum[keys[t]]:
                minimum[keys[t]] = score
                hits[keys[t]] = [score, b[i:i+14], t]
    results1.append(hits)

```

```
results2 = []
for b in base2:
    hits = {}
    minimum = {'BamHI-HindIII':40, 'HindIII-ApaI':40, 'BamHI-
ApaI':40, 'HindIII-BamHI':40, 'ApaI-HindIII':40, 'ApaI-BamHI':40}
    for i,l in enumerate(b[0:-13]):
        for t in targets:
            score = window_score(t, b[i:i+14])
            if score < minimum[keys[t]]:
                minimum[keys[t]] = score
                hits[keys[t]] = [score, b[i:i+14], t]
results2.append(hits)
```